EP 0 736 478 (4) a1

EUROPEAN PATENT OFFICE

Patent Abstracts of Japan

PUBLICATION NUMBER : 09319632
PUBLICATION DATE : 12-12-97

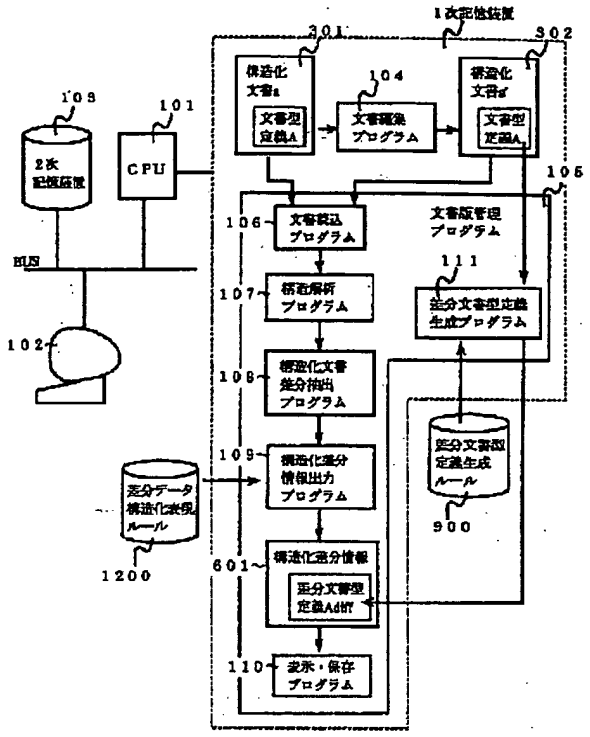
APPLICATION DATE : 31-05-96
APPLICATION NUMBER : 08159202

APPLICANT : HITACHI LTD;

INVENTOR : ITO YASUKI;

INT.CL. : G06F 12/00 G06F 12/00 G06F 17/21

TITLE : METHOD AND DEVICE FOR MANAGING VERSION OF STRUCTURED DOCUMENT



ABSTRACT : PROBLEM TO BE SOLVED: To prevent the capacity of a document database from being make huge in the case of managing the version of a document to be frequently updated by extracting the changed parts of both documents before and after editing, outputting the provided change information as structurally described differential information, displaying and preserving that structured differential information.

SOLUTION: The SGML document of a comparing object is read from a secondary storage device 103 by a document reading program 106. The logical structure of two structured documents to be compared is analyzed by a structure analytic program 107 and based on the provided logical structure information, the difference between structural documents is extracted by a structured document difference extraction program 108. The extracted change information is described in the format of SGML according to differential DTD corresponding to the DTD of the SGML document as a comparing object by a structured differential information output program 109 and outputted as structured differential information. Then, the differential result is displayed on terminal equipment 102 by a display/preservation program 110 and the structured differential data are preserved in the secondary storage device 103.

COPYRIGHT: (C)1997,JPO

EP36478 (4)(b)

(19)日本国特許庁（ＪＰ）　(12)公開特許公報（Ａ）　(11)特許出願公開番号

特開平9－319632

(43)公開日　平成９年(1997)12月12日

(51)Int.Cl.[6]		識別記号	庁内整理番号	ＦＩ		技術表示箇所
Ｇ０６Ｆ	12/00	５１７		Ｃ０６Ｆ	12/00	５１７
		５４７				５４７Ｈ
	17/21				15/20	５７０Ｒ

審査請求　未請求　請求項の数10　ＦＤ　（全 14 頁）

(21)出願番号　特願平8－159202

(22)出願日　平成８年(1996)５月31日

(71)出願人　000005108
株式会社日立製作所
東京都千代田区神田駿河台四丁目６番地

(72)発明者　青山　ゆき
神奈川県川崎市幸区鹿島田890番地の12
株式会社日立製作所情報・通信開発本部内

(72)発明者　高橋　亨
神奈川県川崎市幸区鹿島田890番地の12
株式会社日立製作所情報・通信開発本部内

(72)発明者　東野　純一
神奈川県川崎市幸区鹿島田890番地の12
株式会社日立製作所情報・通信開発本部内

(74)代理人　弁理士　笹岡　茂　（外１名）

最終頁に続く

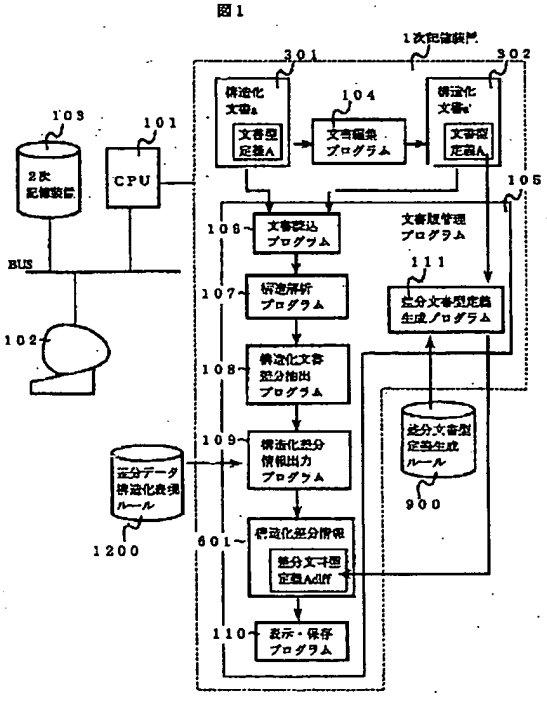
(54)【発明の名称】　構造化文書の版管理方法および装置

(57)【要約】

【課題】　文書の論理構造の変更か文書内容の変更かが簡単に明示でき、SGML記述の構造化文書の特徴を活かした文書編集が行える構造化文書の版管理方法。

【解決手段】　SGML記述の構造化文書に対し編集を施し記憶装置に格納し、記憶装置から編集前後の構造化文書を読み出し、これらの論理的構造を解析し、該論理的構造に基づき構造化文書間の変更箇所を抽出し、該変更箇所を構造化記述した差分情報(構造化差分情報)として出力し、これを表示、保存する。構造化差分情報を記述するための差分文書型定義は構造化文書の文書型定義から生成する。また、複数の版の構造化文書を記憶する際、基準となる文書のみ記憶し、基準以外の文書は構造化差分情報のみを記憶し、指定した版の構造化文書を記憶装置から取り出す際、基準文書はそのまま取り出し、基準以外の文書は、前記構造化差分情報を取り出し、基準文書と該構造化差分情報により指定した版を合成する。

図1



BNSDOCID: <JP_____409319632A__I_>

【特許請求の範囲】

【請求項１】　ＳＧＭＬ(Standard Generalized Markup Language)で記述され文書型定義をされた構造化文書に対して削除、挿入、または変更などの編集を施す処理装置と、該編集前後の構造化文書を格納する記憶装置を備え、前記処理装置により前記編集前後の両構造化文書を管理する構造化文書の版管理方法において、
前記記憶装置から編集前後の構造化文書を読み出す文書読込ステップと、
上記読込ステップで取得された両構造化文書の論理的な構造を解析する構造解析ステップと、
上記構造解析ステップによって得られた論理構造情報に基づいて、上記編集前後の両文書間の変更箇所を抽出する構造化文書差分抽出ステップと、
上記差分抽出ステップにより得られた変更情報を、構造化記述した差分情報として出力する構造化差分情報出力ステップと、
出力された構造化差分情報を表示、保存するステップを有することを特徴とする構造化文書の版管理方法。

【請求項２】　請求項１記載の構造化文書の版管理方法において、
差分抽出する構造化文書の前記文書型定義から、構造化した差分情報のための文書型定義を生成する差分文書型定義生成ステップを有することを特徴とする構造化文書の版管理方法。

【請求項３】　請求項１または請求項２記載の構造化文書の版管理方法において、
前記記憶装置に編集前後の複数の版の構造化文書を記憶する際に、基準となる文書のみ記憶し、基準以外の文書は、基準文書との構造化した差分情報のみを記憶する文書登録ステップを有することを特徴とする構造化文書の版管理方法。

【請求項４】　請求項３記載の構造化文書の版管理方法において、
指定した版の構造化文書を前記記憶装置から取り出す際に、基準となる文書はそのまま取り出し、基準以外の文書は、基準文書と構造化した差分情報を取り出し、指定した版を合成する文書取出ステップを有することを特徴とする構造化文書の版管理方法。

【請求項５】　請求項１記載の構造化文書の版管理方法において、
前記出力された構造化差分情報を表示、保存するステップにおける表示ステップは、
表示画面上に構造を表示する構造ウィンドウと構造中の文字列を表示する文字列ウィンドウを表示し、該構造ウィンドウ中に編集前後の構造の表示と構造の追加、削除等を指示する表示をし、該文字列ウィンドウ中に編集前後の文字列の表示と文字列の追加、削除等を指示する表示を前記構造ウィンドウ中の編集前後の構造の表示の位置と対応する位置に行うことを特徴とする構造化文書の版管理方法。

【請求項６】　記憶装置と処理装置を備え、ＳＧＭＬで記述され文書型定義をされた構造化文書に対して前記処理装置により削除、挿入、または変更などの編集を施し、前記記憶装置に該編集前後の構造化文書を格納し、前記処理装置により前記編集前後の両構造化文書を管理する構造化文書版管理装置において、
前記処理装置は、
前記記憶装置から編集前後の構造化文書を読み出す文書読込手段と、
上記文書読込手段で取得された両構造化文書の論理的な構造を解析する構造化文書解析手段と、
上記構造解析手段によって得られた論理構造情報に基づいて、上記編集前後の構造化文書間の変更箇所を抽出する構造化文書差分抽出手段と、
上記差分抽出手段により得られた変更情報を、構造化記述した差分情報として出力する構造化差分情報出力手段と、
出力された構造化差分情報を表示、保存する手段を備えることを特徴とする構造化文書の版管理装置。

【請求項７】　請求項６記載の構造化文書の版管理装置において、
差分抽出する構造化文書の前記文書型定義から、構造化した差分情報のための文書型定義を生成する差分文書型定義生成手段を備えることを特徴とする構造化文書の版管理装置。

【請求項８】　請求項６または請求項７記載の構造化文書の版管理装置において、
前記記憶装置に編集前後の複数の版の構造化文書を記憶する際に、基準となる文書のみ記憶し、基準以外の文書は、基準文書との構造化した差分情報のみを記憶する文書登録手段を備えることを特徴とする構造化文書の版管理装置。

【請求項９】　請求項８記載の構造化文書の版管理装置において、
前記記憶装置より指定した版の構造化文書を取り出す際に、基準となる文書はそのまま取り出し、基準以外の文書は、基準文書と構造化した差分情報を取り出し、指定した版を合成する文書取出手段を備えることを特徴とする構造化文書の版管理装置。

【請求項１０】　請求項６記載の構造化文書の版管理装置において、
前記出力された構造化差分情報を表示、保存する手段は、
表示画面上に構造を表示する構造ウィンドウと構造中の文字列を表示する文字列ウィンドウを表示し、該構造ウィンドウ中に編集前後の構造の表示と構造の追加、削除等を指示する表示をし、該文字列ウィンドウ中に編集前後の文字列の表示と文字列の追加、削除等を指示する表示を前記構造ウィンドウ中の編集前後の構造の表示の位

BNSDOCID: <JP_____409319632A__I_>

置と対応する位置に行う表示手段を備えることを特徴とする構造化文書の版管理装置。

【発明の詳細な説明】

【０００１】

【発明の属する技術分野】本発明は、電子ファイルとして記憶されている構造化文書を取り扱うワープロ等の文書処理装置における構造化文書の版管理方法及び装置に関する。

【０００２】

【従来の技術】版管理とは、文書を編集、更新していく際に、文書の任意のバージョンを格納し、取り出すことができるよう、各バージョンの文書情報を管理することである。また、保存された各バージョンの文書は、文書の編集を効率よく行うために、バージョン間の変更箇所を抽出して表示したりするためにも用いられる。従来、文書の版管理には二つの方式が採られている。１番目は、各バージョンの文書をすべて保存しておき、指定されたバージョンの文書をそのまま取り出すよう管理する単純版管理方式（従来方式１）である。２番目は、元の文書あるいは最新文書等、基準となるバージョンの文書を一つ保存し、他のバージョンの文書に関しては、基準の文書との差分情報だけ保存しておき、指定されたバージョンの文書を取り出す際は、基準の文書と差分情報から、そのバージョンの文書を合成するよう管理する差分版管理方式（従来方式２）である。また、近年、文書情報の効率的な共有と再利用を目的に、ＳＧＭＬ等を用いて文書を構造化して作成し、利用する動きが活発化してきた。構造化文書は、文書の論理的な構造に関する情報、例えば“文書中のこの部分はタイトルである”、“この部分は章である”といった情報を明確に記述できる。このため、変更や追加が細部に渡り簡単に確実に行えるため、更新が頻繁に行われるマニュアルなどの文書の編集に多用されてきている。

【０００３】この構造化文書の版管理に、従来の二方式を用いると、次のような問題が生じる。まず、従来方式１は、構造化文書を対象とした場合に限らないが、各バージョンの文書をすべて保存しなければならないため、頻繁に更新が発生する場合には、文書データベースの容量が巨大化してしまうという問題が生じる。次に、従来方式２は、文書データベースの容量が巨大化してしまうという従来方式１の問題は回避できるものの、文書の論理構造を区別した差分情報の管理が行えないため、文書のどこの構造が変更されたのか分からず、構造化して文書を作成している利点を活かせず、効率的な文書編集が行えないという問題が生じる。これに対して、「特開平７－２００３７０」（従来方式３）では、構造化文書を対象として各バージョンの文書を保持し、変更箇所を表示する版管理方法が開示されている。本方式では確かに構造情報の差分も差分データとして抽出、表示されるものの、差分の表示方法が一律であるため、構造そのものが変更されたのか、あるいは構造の内容が変更されたのかが分かり難く、効率的な編集が行えないという問題がある。

【０００４】

【発明が解決しようとする課題】編集前後の構造化文書間の変更箇所を抽出した差分データとしては、構造を持たない通常の文書を比較した場合と異なり、次のような特徴がある。

（１）構造自体の変更と構造中の文字列の変更がある。

（２）差分情報にも論理的な構造がある。

図３の構造化文書を例にとり説明する。図３の構造化文書はＳＧＭＬ(Standard Generalized Markup Language)(ISO8879)で記述されたもので、＜Ａ＞と＜／Ａ＞で挟まれた文字列が、論理構造＜Ａ＞に属していることを意味する。この論理構造を表すマークのことをタグと呼び、＜Ａ＞と＜／Ａ＞はそれぞれ開始タグ、終了タグと呼ぶ。さらに、それぞれの構造のことをＥＬＥＭＥＮＴ（エレメント）と呼ぶ。図３（ａ）構造化文書ａを例にとると、＜氏名＞と＜／氏名＞で挟まれた文字列“平成太郎”が、論理構造＜氏名＞に属することを表す。また、＜Ａ＞と＜／Ａ＞の間に＜Ｂ＞と＜／Ｂ＞があることは、論理構造＜Ｂ＞が論理構造＜Ａ＞の子構造であることを意味する。例えば図３（ａ）では、＜発信者＞と＜／発信者＞の間に＜氏名＞と＜／氏名＞があるので、＜氏名＞は＜発信者＞の子構造であることを表す。図３の編集前の構造化文書ａと編集後の構造化文書ａ’を比較した結果の例を図４（ａ）に示す。項番１（４０１）と項番３（４０３）は、それぞれ＜氏名＞，＜本文＞といった文書の構造は変わらず、構造中の文字列が変更されている例である。項番２（４０２）は、＜所属＞という構造自体が新たに挿入されている例である。次に、構造化文書間の差分情報には論理的な構造があるという例を示す。例えば、項番１（４０１）と項番３（４０３）は文字列の変更であるが、これらの文字列の変更箇所が、それぞれ＜氏名＞，＜本文＞という構造の中の変更であるというように、構造を特定して差分情報を表すためには、差分データとして構造情報を持たなければならない。また、項番２（４０２）では、挿入された＜所属＞は＜発信者＞の子構造であるという、構造情報を持っている。

【０００５】しかしながら、従来の方式では、このような構造化文書の差分データの特徴を効果的に表示することができなかった。従来方式３では、その変更箇所が文書の論理的な構造に関する情報の変更であっても、文字列の変更と区別することなく同様に構造を表す文字の表示属性を変えて表示している。このため、構造そのものが変更されたのか、あるいは構造の内容が変更されたのかがユーザにとって分かり難いという問題がある。この問題を具体的な例で説明する。図４（ｂ）に従来方式３による差分データの表示の例を示す。本図では、構造自

BNSDOCID: <JP_____409319632A_I_>

体の変更と構造中の文字列の変更を区別せず、構造情報を無視した表示方法になっている。このため、文書編集ソフト等でこの構造化文書を編集しているユーザにとって、どんな変更が行われたのか分かり難い。また、文書編集ソフト等が、構造化文書を表示する際に構造情報をｔｒｅｅで表すなど専用の表示プログラムを用いる場合、図４のような差分データの表示には、別の表示プログラムが必要になるためプログラムが複雑になってしまう。

【０００６】また、文書の任意のバージョンを格納し、取り出す機能を実現するために、従来方式１のように各バージョンの文書をすべて保存するのでは、保存するデータ量が多くなり、頻繁にバージョンが更新される文書の版管理方式としては適さない。そこで、改訂が頻繁に行なわれる構造化文書の版管理方式としては、元の文書あるいは最新文書等、基準となるバージョンの文書を一つ保存し、他のバージョンの文書に関しては、基準の文書との差分情報だけ保存しておき、指定されたバージョンの文書を取り出す際は、基準の文書と差分情報から、そのバージョンの文書を合成する従来方式２のような方式が用いられるが、この方式では、構造化文書を対象とした文書の論理構造を区別した比較に基づく、版管理は行えないという問題がある。

【０００７】本発明の目的は、これらの問題に対して、ＳＧＭＬ等を用いて記述される文書などのように、頻繁に更新が発生する文書の版を管理する際、文書データベースの容量の巨大化を防ぐとともに、文書の論理構造を区別した差分の管理が行え、さらに論理構造そのものの変更かその内容の変更かが簡単に明示でき、構造化文書の特徴を活かした効率的な文書編集が行えるようにする、構造化文書の版管理方法を提供することにある。

【０００８】

【課題を解決するための手段】上記目的を達成するため、本発明は、ＳＧＭＬ(Standard Generalized Markup Language)で記述され文書型定義をされた構造化文書に対して削除、挿入、または変更などの編集を施す処理装置と、該編集前後の構造化文書を格納する記憶装置を備え、前記処理装置により前記編集前後の両構造化文書を管理する構造化文書の版管理方法において、前記記憶装置から編集前後の構造化文書を読み出す文書読込ステップと、上記読込ステップで取得された両構造化文書の論理的な構造を解析する構造解析ステップと、上記構造解析ステップによって得られた論理構造情報に基づいて、上記編集前後の両文書間の変更箇所を抽出する構造化文書差分抽出ステップと、上記差分抽出ステップにより得られた変更情報を、構造化記述した差分情報として出力する構造化差分情報出力ステップと、出力された構造化差分情報を表示、保存するステップを有するようにしている。さらに、差分抽出する構造化文書の前記文書型定義から、構造化した差分情報のための文書型定義を生成する差分文書型定義生成ステップを有するようにしている。さらに、前記記憶装置に編集前後の複数の版の構造化文書を記憶する際に、基準となる文書のみ記憶し、基準以外の文書は、基準文書との構造化した差分情報のみを記憶する文書登録ステップを有するようにしている。さらに、指定した版の構造化文書を前記記憶装置から取り出す際に、基準となる文書はそのまま取り出し、基準以外の文書は、基準文書と構造化した差分情報を取り出し、指定した版を合成する文書取出ステップを有するようにしている。さらに、前記出力された構造化差分情報を表示、保存するステップにおける表示ステップは、表示画面上に構造を表示する構造ウィンドウと構造中の文字列を表示する文字列ウィンドウを表示し、該構造ウィンドウ中に編集前後の構造の表示と構造の追加、削除等を指示する表示をし、該文字列ウィンドウ中に編集前後の文字列の表示と文字列の追加、削除等を指示する表示を前記構造ウィンドウ中の編集前後の構造の表示の位置と対応する位置に行うようにしている。

【０００９】また、記憶装置と処理装置を備え、ＳＧＭＬで記述され文書型定義をされた構造化文書に対して前記処理装置により削除、挿入、または変更などの編集を施し、前記記憶装置に該編集前後の構造化文書を格納し、前記処理装置により前記編集前後の両構造化文書を管理する構造化文書版管理装置において、前記処理装置は、前記記憶装置から編集前後の構造化文書を読み出す文書読込手段と、上記文書読込手段で取得された両構造化文書の論理的な構造を解析する構造化文書解析手段と、上記構造解析手段によって得られた論理構造情報に基づいて、上記編集前後の構造化文書間の変更箇所を抽出する構造化文書差分抽出手段と、上記差分抽出手段により得られた変更情報を、構造化記述した差分情報として出力する構造化差分情報出力手段と、出力された構造化差分情報を表示、保存する手段を備えるようにしている。さらに、差分抽出する構造化文書の前記文書型定義から、構造化した差分情報のための文書型定義を生成する差分文書型定義生成手段を備えるようにしている。さらに、前記記憶装置に編集前後の複数の版の構造化文書を記憶する際に、基準となる文書のみ記憶し、基準以外の文書は、基準文書との構造化した差分情報のみを記憶する文書登録手段を備えるようにしている。さらに、前記記憶装置より指定した版の構造化文書を取り出す際に、基準となる文書はそのまま取り出し、基準以外の文書は、基準文書と構造化した差分情報を取り出し、指定した版を合成する文書取出手段を備えるようにしている。さらに、前記出力された構造化差分情報を表示、保存する手段は、表示画面上に構造を表示する構造ウィンドウと構造中の文字列を表示する文字列ウィンドウを表示し、該構造ウィンドウ中に編集前後の構造の表示と構造の追加、削除等を指示する表示をし、該文字列ウィンドウ中に編集前後の文字列の表示と文字列の追加、削除

BNSDOCID: <JP_____409319632A_I_>

等を指示する表示を前記構造ウィンドウ中の編集前後の構造の表示の位置と対応する位置に行う表示手段を備えるようにしている。

【００１０】

【発明の実施の形態】以下、本発明の実施の形態の例を説明する。

【００１１】《実施例１》一番目の実施例の構成を図１に示す。図示したように、本実施例は、それぞれ、ＣＰＵ１０１、端末装置１０２、文書を記憶するための２次記憶装置１０３と、文書の編集を行う文書編集プログラム１０４、編集前後の文書を管理する文書版管理プログラム１０５で構成され、さらに文書版管理プログラム１０５は、２次記憶装置１０３から編集前後の構造化文書を読み出す文書読込プログラム１０６、読み込まれた両構造化文書の論理構造を解析する構造解析プログラム１０７、得られた論理構造情報に基づいて、編集前後の両文書間の変更箇所を抽出する構造化文書差分抽出プログラム１０８、変更情報を、構造化記述した差分情報として出力する構造化差分情報出力プログラム１０９、出力された構造化差分情報を表示、保存する表示・保存プログラム１１０、比較する構造化文書の文書型定義から、差分データ用の文書型定義を生成する差分文書型定義生成プログラム１１１から構成される。

【００１２】本実施例では、構造化文書としてＳＧＭＬ文書を例にしている。ＳＧＭＬは、マーク付けされた構造化文書としてＩＳＯの国際規格として定められた文書記述言語のことである。また、ＳＧＭＬ文書はＤＴＤ（文書型定義）によって、その論理構造が予め定義される。ＳＧＭＬ文書はＤＴＤに定義された論理構造に従うよう実際の文書の中身である文書インスタンスが記述される。すなわち、図５に示すように、文書インスタンスと文書型定義をあわせて、構造化文書となる。

【００１３】本実施例の具体的な処理手順を、図２のフローチャートを用いて説明する。その後、処理手順に従って、処理例を説明する。

〈ステップ２０１〉文書編集プログラム１０４で、構造化文書の編集を行う。

【００１４】〈ステップ２０２〉編集前後の構造化文書間の変更箇所を表示したり、保存したりするために、文書版管理プログラム１０５が呼び出されたら、まず、比較対象であるＳＧＭＬ文書のＤＴＤに対応した差分データ用のＤＴＤを取得する。この差分データ用のＤＴＤは、後述するステップ２０６において構造化文書間の変更情報をＳＧＭＬ形式で記述するための文書型定義として用いられる。対応する差分ＤＴＤが存在する場合は、これを２次記憶装置１０３から読み込み、存在しない場合は、差分文書型定義生成プログラム１１１で対応する差分ＤＴＤを生成する。

〈ステップ２０３〉比較対象であるＳＧＭＬ文書を２次記憶装置１０３から文書読込プログラム１０６によって読み込む。

〈ステップ２０４〉差分抽出の前処理として、比較する二つの構造化文書の論理構造を構造解析プログラム１０７によって解析する。

〈ステップ２０５〉ステップ２０４で得られた論理構造情報に基づいて、構造化文書間の差分を構造化文書差分抽出プログラム１０８によって抽出する。

〈ステップ２０６〉構造化差分情報出力プログラム１０９により、ステップ２０５で抽出された変更情報を、ステップ２０２で取得した差分ＤＴＤに従いＳＧＭＬ形式で記述し、構造化差分情報として出力する。

〈ステップ２０７〉表示・保存プログラム１１０により、端末装置１０２に差分結果の表示を行い、また、２次記憶装置１０３に構造化差分データを保存する。

【００１５】（処理例）実施例の具体的な処理例として、図３の構造化文書を例にとり説明する。本処理例では、図３の構造化文書ａ（３０１）と構造化文書ａ'（３０２）の差分を抽出して、図６のようなＳＧＭＬ形式で記述された構造化差分データを出力し、その結果を表示、保存することを目的としている。構造化文書の差分データとしては、構造自体の変更と構造中の文字列の変更の場合があるため、図６では、次のように差分の構造情報を記述している。すなわち、構造自体の変更は、その構造を示すタグにｄｉｆｆｆｌａｇという属性を持たせて構造の変更を表現し、構造中の文字列の変更は、差分を表すタグでその文字列を挟んで表現している。図６の例では、“＜所属＞ＡＢＣ会社＜／所属＞”という構造自体の挿入を“＜所属　ｄｉｆｆｆｌａｇ＝挿入＞ＡＢＣ会社＜／所属＞”というＳＧＭＬ形式で記述している。また、構造中の文字列の変更は、＜挿入＞，＜変更前＞，＜変更後＞などの差分を表すタグでその文字列を挟んで記述している。このｄｉｆｆｆｌａｇという属性名やその属性値、および文字列の差分を表すタグ等は、任意に決めることができる。

【００１６】図２のフローチャートに従って、処理例を説明する。

〈ステップ２０１〉文書編集プログラム１０４で、構造化文書の編集を行う。図３（ａ）の構造化文書ａ（３０１）から図３（ｂ）の構造化文書ａ'（３０２）を編集したとする。

〈ステップ２０２〉編集前後の構造化文書間の変更箇所を表示したり、保存したりするために、文書版管理プログラム１０５が呼び出されたら、比較対象であるＳＧＭＬ文書のＤＴＤ（文書型定義）に対応した、差分ＤＴＤを読み込む。存在しない場合は、差分文書型定義生成プログラム１１１で対応する差分ＤＴＤを生成する。

【００１７】例えば、図３の構造化文書は図７のようなＤＴＤを持つ。すなわち、図３の構造化文書は、図７のＤＴＤの定義に従って書かれている。図７のＤＴＤは、それぞれ次のような意味を定義している。まず、７０１

BNSDOCID: <JP_____409319632A_I_>

の一行は、<メモ>という構造は、<発信者>および<本文>という、二つの子構造を持つことを定義している。７０２の一行は、<発信者>という構造は、<氏名>および<所属>という、二つの子構造を持つことを定義している。また、“所属”の後ろにある“？”は出現指示子の一つで、<所属>という構造は文書中に現われる回数が０回または１回であるという意味を表す（“？”以外にも出現指示子として“＊”と“＋”があり、それぞれ出現指標子の付いている構造の現れる回数が０回以上および１回以上という意味を表す）。７０３、７０４、７０５はそれぞれ、<氏名>，<所属>，<本文>という構造は文字列データ（＃ＰＣＤＡＴＡ）を持つことを定義している。

【００１８】この比較する構造化文書のＤＴＤから、差分文書型定義生成プログラム１１１で対応する差分ＤＴＤを生成する。この差分ＤＴＤは、元のＤＴＤに対して、差分文書型定義生成ルールを適用することにより生成される。差分文書型定義生成ルールの例を図９に示す。図７のＤＴＤに対し、図９の差分文書型定義生成ルール（９００）を適用すると図８の差分ＤＴＤが生成される。

【００１９】図９の差分文書型定義生成ルールは次のようなルールで構成される。項番１（９０１）、２（９０２）のルールは、元のＤＴＤでは文字データとして定義されている箇所に、文字列の変更情報として<挿入>、<削除>、<変更前>、<変更後>という差分を表すタグが挿入できるように、ＤＴＤを変更している。すなわち、構造中の文字列が変更された時に、その文字列を<挿入>、<削除>、<変更前>、<変更後>というタグで挟むための定義である。項番３（９０３）のルールは、元のＤＴＤで定義されている構造のうち、最上位の構造以外には、その構造の属性としてｄｉｆｆｆｌａｇを持つよう、ＤＴＤを変更している。すなわち、構造自体が変更された時に、その構造に差分を表す属性をつけるための定義である。ｄｉｆｆｆｌａｇ属性は、“ＮＵＬＬ”、“挿入”、“削除”、“変更前”、“変更後”いずれかの値を持つ。また、ｄｉｆｆｆｌａｇが省略された場合は、構造の変更がなかったという意味で、“ＮＵＬＬ”という値が与えられる。項番４（９０４）のルールは、元のＤＴＤで定義されている構造の出現指示子を変更する。出現指示子はその構造の出現回数を表すもので、“？”は現れる回数が０回または１回であること、“＊”は現れる回数が０回または１回以上であること、“＋”は現れる回数が１回以上であること、出現指示子がない場合は現れる回数が必ず１回であることを意味している。差分ＤＴＤでは、変更のなかった構造は差分データに含めないことが可能となるよう、構造が現れる回数が０回であることを許すように、ＤＴＤを変更している。

【００２０】図７のＤＴＤに対し、この差分文書型定義生成ルール（９００）を適用すると図８の差分ＤＴＤが生成される。例えば、項番１（９０１）のルールにより、図８の８０１が挿入される。項番２（９０２）のルールにより、図７の７０３，７０４，７０５が図８の８０４，８０５，８０６に置き換えられる。項番３（９０３）のルールにより、図８の８０７が挿入される。項番４（９０４）のルールにより、図７の７０１，７０２の出現指示子がそれぞれ置き換えられ、図８の８０２，８０３となる。このように図９のルールを用いて生成した、図８の差分ＤＴＤは、図６の差分データを表す文書型定義となっている。

【００２１】〈ステップ２０３〉比較対象である図３（ａ）の構造化文書ａ（３０１）と図３（ｂ）の構造化文書ａ’（３０２）を２次記憶装置１０３から文書読込プログラム１０６によって読み込む。

〈ステップ２０４〉構造化文書の論理構造を構造解析プログラム１０７によって解析する。図３の構造化文書（ａ）および（ｂ）を解析すると、図１０の文書木（ａ）と（ｂ）が得られる。文書木とは、文書の論理構造を表す木構造のことで、木構造の根には、ＳＧＭＬ文書の最上位ＥＬＥＭＥＮＴが割り当てられ、木構造の末端に構造中の文字列が割り当てられる。

【００２２】〈ステップ２０５〉ステップ２０４で得られた論理構造情報に基づいて、構造化文書間の差分を構造化文書差分抽出プログラム１０８によって抽出する。構造化文書間の差分抽出は次のように行なう。まず、ステップ２０４で得られた文書木のノードを単位に差分を抽出する。これは、文書木のノードは文書の構造単位になっているため、構造を単位に差分を抽出することと等価である。文書木間で同じ構造名や文字列を持つノードは、一致しているとして対応づける。次に、対応づけられなかったノードを今度は文字を単位に差分抽出する。例えば、図１０の文書木（ａ）と（ｂ）のノードを単位に差分を抽出すると、“<メモ>”，“<発信者>”，“<氏名>”，“<本文>”が一致しているとして対応づけられる。次に、対応づけられなかったノードを今度は文字を単位に差分抽出する。“平成太郎”と“昭和次郎”を文字単位で差分抽出すると一致する文字はないので、構造中の文字列の変更箇所として抽出される。“<所属>”および“ＡＢＣ会社”は挿入として抽出される。“<所属>”は構造を表すノードなので、構造自体の変更として抽出される。また、“こんにちは。”と“こんにちは。お元気ですか？”を文字単位で差分抽出すると“お元気ですか？”が文字列の変更箇所として抽出される。その結果、図１１ような論理構造をもった差分データが得られる。

【００２３】〈ステップ２０６〉構造化差分情報出力プログラム１０９により、ステップ２０５で抽出された変更情報を、ステップ２０２で取得した差分ＤＴＤに従いＳＧＭＬ形式で記述し、構造化差分情報として出力す

BNSDOCID: <JP_____409319632A_I_>

る。図8の差分ＤＴＤに従い、図１１の変更情報をＳＧＭＬ形式で出力すると、図6の構造化差分データが得られる。この構造化差分データは、変更情報に対して、差分データの構造化表現ルールを適用することで生成される。差分データの構造化表現ルールの例を図１２に示す。１２０１が構造自体の変更を、１２０２が構造中の文字列の変更を記述するためのルールの例である。すなわち、構造自体の変更は、その構造を示すタグにｄｉｆｆｆｌａｇという属性を持たせて構造の変更を表現し、構造中の文字列の変更は、＜挿入＞，＜変更前＞，＜変更後＞などの差分を表すタグでその文字列を挟んで表現する。このｄｉｆｆｆｌａｇという属性名やその属性値、および文字列の差分を表すタグ等は、任意に決めることができる。

【００２４】〈ステップ２０７〉表示・保存プログラム１１０により、端末装置１０２に差分結果の表示を行い、２次記憶装置１０３に構造化差分データを保存する。差分データがＳＧＭＬ形式で出力されているため、差分データの表示にＳＧＭＬ専用のエディタや、ビューアを使って、そのまま表示することができる。図１３にＳＧＭＬ専用エディタを使った、構造化文書の表示例および、図１４に差分データの表示例を示す。図１３の１３０１は構造を表示するウィンドウで、１３０２がその構造中の文字列を表示するウィンドウである。図１４では、図6の差分データを構造化表示している例である。この際、構造自体の変更箇所は、構造を表すマークの色を変える、種類を変える、太線で囲む等、構造の表示を他とは区別して表示する。また、文字列の変更箇所は、同様に、他の文字列と区別して表示する。

【００２５】以上のステップにより、構造化文書間の変更箇所を抽出し、構造化差分データとして出力することが可能となる。これにより、本方式を文書比較機能としてＳＧＭＬ文書編集ソフトに組み込むことで、差分データを直接構造化表示することが可能となり、例えば、構造自体の変更と構造中の文字列の変更を区別したりすることで、文書編集ソフト等でこの構造化文書を編集しているユーザにとって、どんな変更が行われたのか分かり易くすることができる。また、文書編集ソフト等が、構造化文書を表示する際に構造情報をｔｒｅｅで表すなど専用の表示プログラムを用いる場合でも、別の表示プログラムを必要とせず変更箇所を表示することが可能となる。

【００２６】《実施例２》二番目の実施例の構成を図１５に示す。図示したように、本実施例は、それぞれ、ＣＰＵ１５０１、端末装置１５０２、文書を記憶するための２次記憶装置１５０３と、文書の版管理を行う文書版管理プログラム１５０４で構成され、さらに文書版管理プログラム１５０４は、文書を２次記憶装置１５０３から取り出す文書取出プログラム１５０５、文書を２次記憶装置１５０３に登録する文書登録プログラム１５０６、構造化文書間の変更箇所を論理構造情報に基づいて抽出し、構造化差分データとして出力する差分抽出プログラム１５０７、出力された構造化差分データを表示、編集する表示・編集プログラム１５０８から構成される。

【００２７】本実施例の具体的な処理手順を、図１６のフローチャートを用いて説明する。

〈ステップ１６０１〉表示・編集プログラム１５０８により、構造化文書を編集する。

〈ステップ１６０２〉編集した構造化文書を登録するために、文書版管理プログラム１５０４が呼び出されたら、まず、その文書が新規文書であれば、文書登録プログラム１５０６により２次記憶装置１５０３に文書全体を保存し、バージョン１．０（Ｖ１．０）として登録する。

〈ステップ１６０３〉新規文書でなければ、２次記憶装置１５０３に登録されているＶ１．０の文書を文書取出プログラム１５０５により読み込む。Ｖ１．０文書と登録する文書を差分抽出プログラム１５０７により比較し、ＳＧＭＬ形式で記述した構造化差分データを生成する。この構造化差分データの生成には、実施例１の方式を用いる。

【００２８】〈ステップ１６０４〉ステップ１６０３で生成された構造化差分データを、文書登録プログラム１５０６により２次記憶装置１５０３に新規バージョンとして登録する。

〈ステップ１６０５〉任意のバージョンの文書を表示、編集するために、文書取出プログラム１５０５が呼び出された場合、Ｖ１．０の文書の取出しなら、２次記憶装置１５０３からＶ１．０の文書を読み込む。

〈ステップ１６０６〉Ｖ１．０以外の文書の取出しなら、文書取出プログラム１５０５により２次記憶装置１５０３から、Ｖ１．０の文書とその指定されたバージョンの差分データを読み込み、そのバージョンの文書を合成する。

〈ステップ１６０７〉編集が終了していなければ、ステップ１６０１へ戻る。

【００２９】（処理例）実施例の具体的な処理例として、図１７の構造化文書を例にとり説明する。表示・編集プログラム１５０８により、図１７（ａ）の構造化文書ｂ（Ｖ１．０）を作成し、文書登録プログラム１５０６により２次記憶装置１５０３に文書をＶ１．０として登録したとする。次に構造化文書ｂを編集するために、文書取出プログラム１５０５によりＶ１．０を読み込む。構造化文書ｂを編集して、図１７（ｂ）の構造化文書ｂ’に変更する。この文書をバージョン２．０（Ｖ２．０）として登録するために文書登録プログラム１５０６を呼び出す。新規文書ではないので、２次記憶装置１５０３に登録されているＶ１．０の文書を取り出し、差分抽出プログラム１５０７によりＶ１．０の文書とＶ

BNSDOCID: <JP_____409319632A__I_>

２．０の文書を比較し、図１８の構造化差分データを生成する。生成された構造化差分データを、文書登録プログラム１５０６により登録する。

【００３０】Ｖ２．０の取出しが指定された場合は、文書取出プログラム１５０５により、まず、図１７（ａ）Ｖ１．０と図１８のＶ２．０に対応する構造化差分データを読み込む。これらから、Ｖ２．０の文書を合成する。合成は次のように行う。まず、Ｖ１．０の文書のうち、構造化差分データに対応する構造を差分データで置き換える。置き換えたものを図１９の１９０１に示す。次に、＜変更前＞，＜削除＞タグで挟まれた文字列、およびｄｉｆｆｆｌａｇ属性値が“変更前”，“削除”の構造は削除する。さらに、差分を表すタグや属性をすべて削除する。これにより、図１９の１９０２ようなＶ２．０の文書が再現される。

【００３１】なお本実施例では、Ｖ１．０の文書を基準文書としてバージョン管理を行なっているが、最新文書を基準文書としても、同様の方式でバージョン管理することができる。以上の方式により、基準となるバージョンの文書を一つ保存し、他のバージョンの文書に関しては、基準の文書との差分情報のみ保存するだけで、文書の任意のバージョンを取り出す機能が実現が容易に実現できる。本方式は、長大な文書の一部を変更した場合などに、バージョン管理のデータ量が少なくなるため、特に有効な方式であることが分かる。

【００３２】

【発明の効果】本発明によれば、編集前後の文書間の変更箇所を構造化された差分データとして抽出することができ、差分データをＳＧＭＬ文書編集ソフト等で直接構造化表示することが可能となる。これにより、構造自体の変更と構造中の文字列の変更を区別して表示したりすることができ、文書編集ソフト等でこの構造化文書を編集しているユーザにとって、どんな変更が行われたのか分かり易くすることが可能となるため、構造化文書の編集作業の効率が上がる。また、文書編集ソフト等が、構造化文書を表示する際に構造情報をｔｒｅｅで表すなど専用の表示プログラムを用いる場合でも、別の表示プログラムを必要とせず変更箇所を表示することができるため、構造化文書専用の文書編集ソフト等に文書比較機能を組み込むことが容易に実現できる。また、本発明によって出力された構造化差分データは、ＳＧＭＬ専用のビューアを使って直接構造化表示することができるので、新たに改訂履歴ビューア等を作成することなく、改訂履歴データを閲覧することが可能となる。さらに、本発明によれば、基準となるバージョンの文書を一つ保存し、他のバージョンの文書に関しては、基準の文書との差分情報を保存するだけですむため、文書の任意のバージョンを取り出す機能が容易に実現でき、長大な文書の一部を変更した場合や文書を頻繁に変更した場合など、バージョン管理のデータ量を少なくすることが可能となる。

【図面の簡単な説明】

【図１】本発明の第一の実施例の構成図である。

【図２】本発明の第一の実施例の処理手順を示す図である。

【図３】第一の実施例を説明するための構造化文書の第一の例を示す図である。

【図４】構造化文書の第一の例を差分抽出した差分データ例、および従来の方式により差分データを表示した表示例を示す図である。

【図５】構造化文書と文書型定義の関係を説明する図である。

【図６】構造化文書の第一の例を構造化差分抽出して出力した構造化差分データ例を示す図である。

【図７】構造化文書の第一の例の文書型定義の例を示す図である。

【図８】構造化文書の第一の例の文書型定義より生成した差分文書型定義の例を示す図である。

【図９】構造化文書の文書型定義より差分文書型定義を生成するためのルールを示す図である。

【図１０】構造化文書の第一の例から作成した文書木を示す図である。

【図１１】構造化文書の第一の例を構造化差分抽出した結果の文書木を示す図である。

【図１２】構造化差分抽出した差分データを構造化差分データとして出力するための構造化表現ルールの例を示す図である。

【図１３】構造化文書の第一の例の表示例である。

【図１４】構造化文書の第一の例の構造化差分データの表示例である。

【図１５】本発明の第二の実施例の構成図である。

【図１６】本発明の第二の実施例の処理手順を示す図である。

【図１７】第二の実施例を説明するための構造化文書の第二の例を示す図である。

【図１８】構造化文書の第二の例の構造化差分データ例を示す図である。

【図１９】構造化文書の第二の例の基準文書と構造化差分データを合成する手法を説明するための図である。

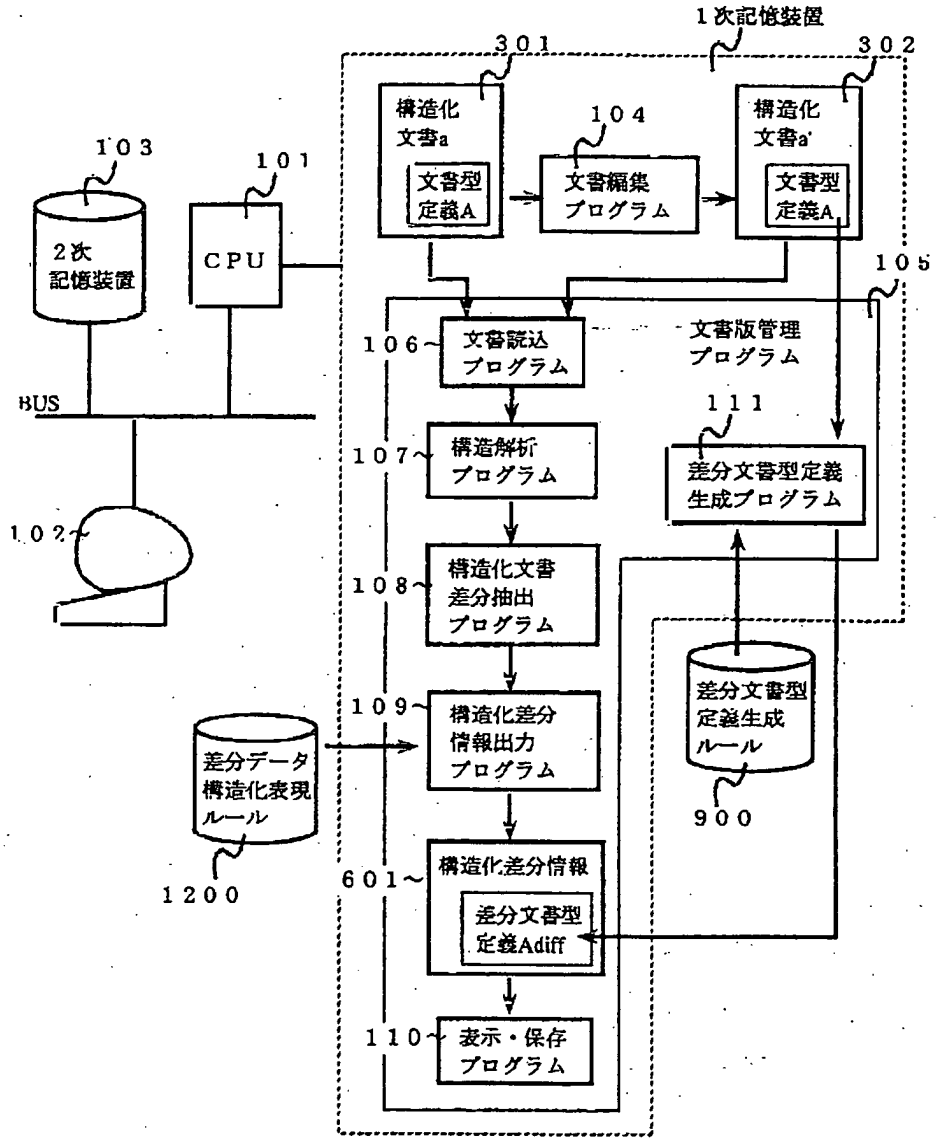
【符号の説明】

１０１　ＣＰＵ
１０２　端末装置
１０３　２次記憶装置
１０４　文書編集プログラム
１０５　文書版管理プログラム
１０６　文書読込プログラム
１０７　構造解析プログラム
１０８　構造化文書差分抽出プログラム
１０９　構造化差分情報出力プログラム
１１０　文書表示・保存プログラム
１１１　差分文書型定義生成プログラム

NSDOCID: <JP_____409319632A_I_>

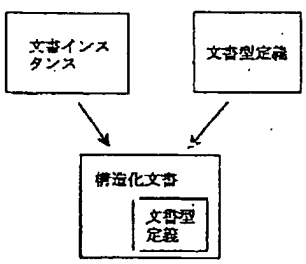
【図1】

図1



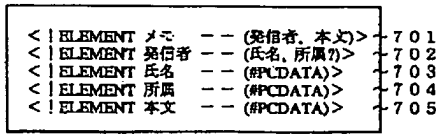
【図5】

図5



【図7】

図7

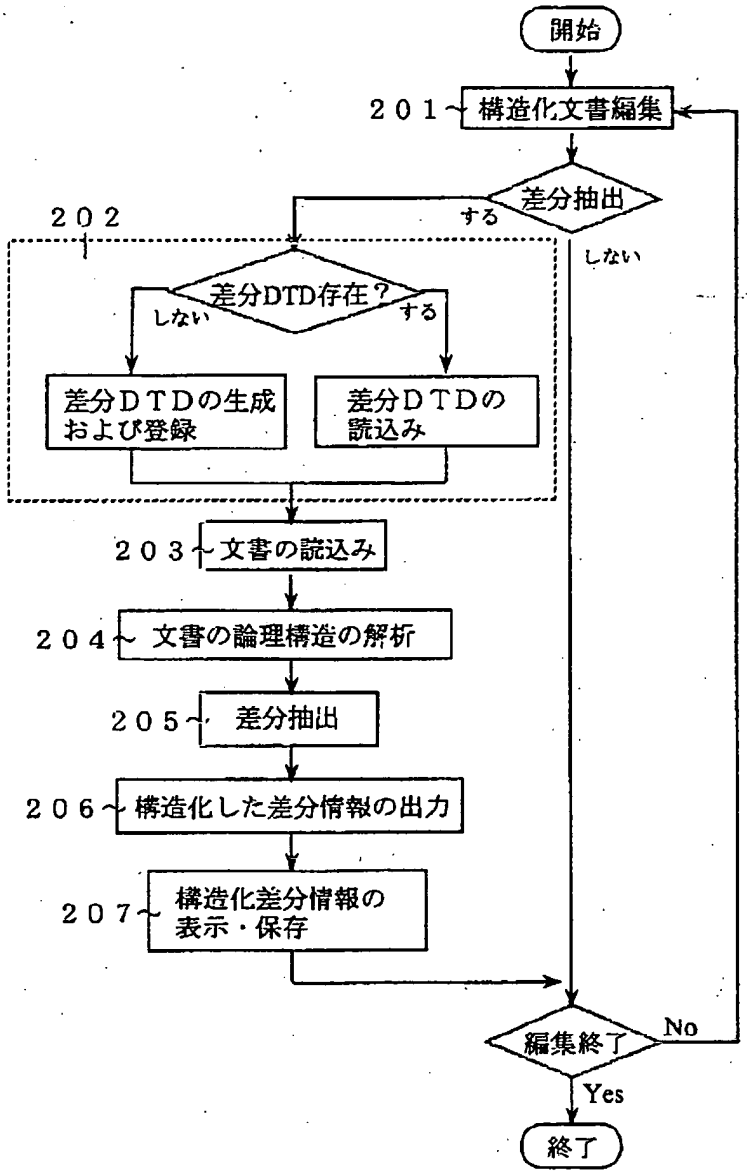


DTD(文書型定義)の例

BNSDOCID: <JP_____409319632A_I_>

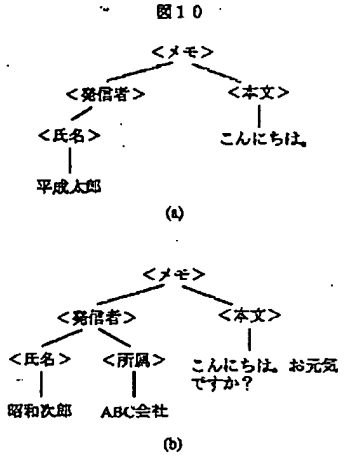
【図2】

図2



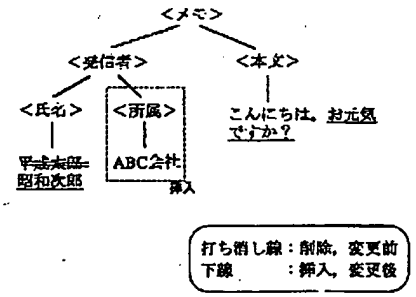
【図10】

図10



【図11】

図11



BNSDOCID: <JP_____409319632A_I_>

【図3】

図3

```
<メモ>
  <発信者>
    <氏名>平成太郎</氏名>
  </発信者>
  <本文>
    こんにちは。
  </本文>
</メモ>
```

301

(a) 構造化文書a

```
<メモ>
  <発信者>
    <氏名>昭和次郎</氏名>
    <所属>ABC会社</所属>
  </発信者>
  <本文>
    こんにちは。お元気ですか？
  </本文>
</メモ>
```

302

(b) 構造化文書a'

【図4】

図4

項番	差分データ
1	<氏名>中"平成太郎"が"昭和次郎"に変更
2	<発信者>の子構造の"<所属>ABC会社</所属>"が構造<氏名>の後ろに挿入
3	<本文>中"お元気ですか？"が7文字目に挿入

401　402　403

(a) 差分データ例

変更箇所

```
<メモ>
  <発信者>
    <氏名>昭和次郎</氏名>
    <所属>ABC会社</所属>
  </発信者>
  <本文>
    こんにちは。お元気ですか？
  </本文>
</メモ>
```

(b) 従来方式による
差分データの表示例

【図6】

図6

```
<メモ>
  <発信者>
    <氏名>
      <変更前>平成太郎</変更前>
      <変更後>昭和次郎</変更後>
    </氏名>
    <所属 diffflag=挿入>ABC会社</所属>
  </発信者>
  <本文>
    こんにちは。<挿入>お元気ですか？</挿入>
  </本文>
</メモ>
```

601

構造化差分データ例

【図8】

図8

```
<!ENTITY % DiffElement "挿入 | 削除 | 変更前 | 変更後">
<!ELEMENT 挿入   -- (#PCDATA)>
<!ELEMENT 削除   -- (#PCDATA)>
<!ELEMENT 変更前 -- (#PCDATA)>
<!ELEMENT 変更後 -- (#PCDATA)>

<!ELEMENT メモ   -- (発信者?, 本文?)>
<!ELEMENT 発信者 -- (氏名?, 所属?)>
<!ELEMENT 氏名   -- (#PCDATA | %DiffElement;)* >
<!ELEMENT 所属   -- (#PCDATA | %DiffElement;)* >
<!ELEMENT 本文   -- (#PCDATA | %DiffElement;)* >

<!ATTLIST 発信者 diffflag
          (NULL | 挿入 | 削除 | 変更前 | 変更後)  NULL>
<!ATTLIST 氏名   diffflag
          (NULL | 挿入 | 削除 | 変更前 | 変更後)  NULL>
<!ATTLIST 所属   diffflag
          (NULL | 挿入 | 削除 | 変更前 | 変更後)  NULL>
<!ATTLIST 本文   diffflag
          (NULL | 挿入 | 削除 | 変更前 | 変更後)  NULL>
```

801　802　803　804　805　806　807

差分DTD(文書型定義)の例

【図18】

図18

```
<論文>
  <著者名>
    <変更前>平成太郎</変更前>
    <変更後>昭和次郎</変更後>
  </著者名>
  <所属 diffflag="挿入">XYZ大学</所属>
  <章>
    <章番号>第1章</章番号>
    構造化文書とは？<削除>SGMLとは？</削除>
  </章>
</論文>
```

1801

BNSDOCID: <JP_____409319632A__I_>

【図9】

図9

900

項番	差分DTD生成ルール	
1	<！ENTITY % DiffElement "挿入｜削除｜変更前｜変更後"> <！ELEMENT 挿入 －－ (#PCDATA)> <！ELEMENT 削除 －－ (#PCDATA)> <！ELEMENT 変更前 －－ (#PCDATA)> <！ELEMENT 変更後 －－ (#PCDATA)> をDTDの先頭に挿入	～901
2	文字列データ(#PCDATA, CDATA等)は、DiffElementを含むよう置き換える。 ex. #PCDATA → (#PCDATA｜%DiffElement;)*	～902
3	最上位ELEMENT以外は、属性diffflagを加える。 diffflagは属性値として、"NULL"，挿入"，"削除"，"変更前"，"変更後"のいずれかの値を持つよう定義する。 属性の省略値を"NULL"とする。	～903
4	出現指示子はそれぞれ次のように置き換える。 なし，？ → ？ ＋，＊ → ＊	～904

差分DTD生成ルールの例

【図12】

図12

1200

差分データの種類		構造化表現ルール
構造自体の差分	挿入	その構造の属性diffflagの値を"挿入"とする
	削除	その構造の属性diffflagの値を"削除"とする
	変更	変更前後の構造の属性diffflagの値をそれぞれ"変更前"および"変更後"とする
構造中の文字の差分	挿入	差分文字列を＜挿入＞タグで挟む
	削除	差分文字列を＜削除＞タグで挟む
	変更	変更前の文字列を＜変更前＞タグで，変更後の文字列を＜変更後＞タグで挟む

1201～　1202～

構造化差分データの構造化表現ルール例

【図13】

図13

構造化文書a.SGM

メモ
発信者
氏名　平成太郎
本文　こんにちは。

1301　(a)　1302

構造化文書a'.SGM

メモ
発信者
氏名　軸和次郎
所属　ＡＢＣ会社
本文　こんにちは。お元気ですか？

(b)

【図14】

図14

構造化文書の差分データ.SGM

メモ
発信者
氏名　平成太郎→軸和次郎
所属　ＡＢＣ会社
本文　こんにちは。お元気ですか？

1401

【図15】

図15

1次記憶装置　1504

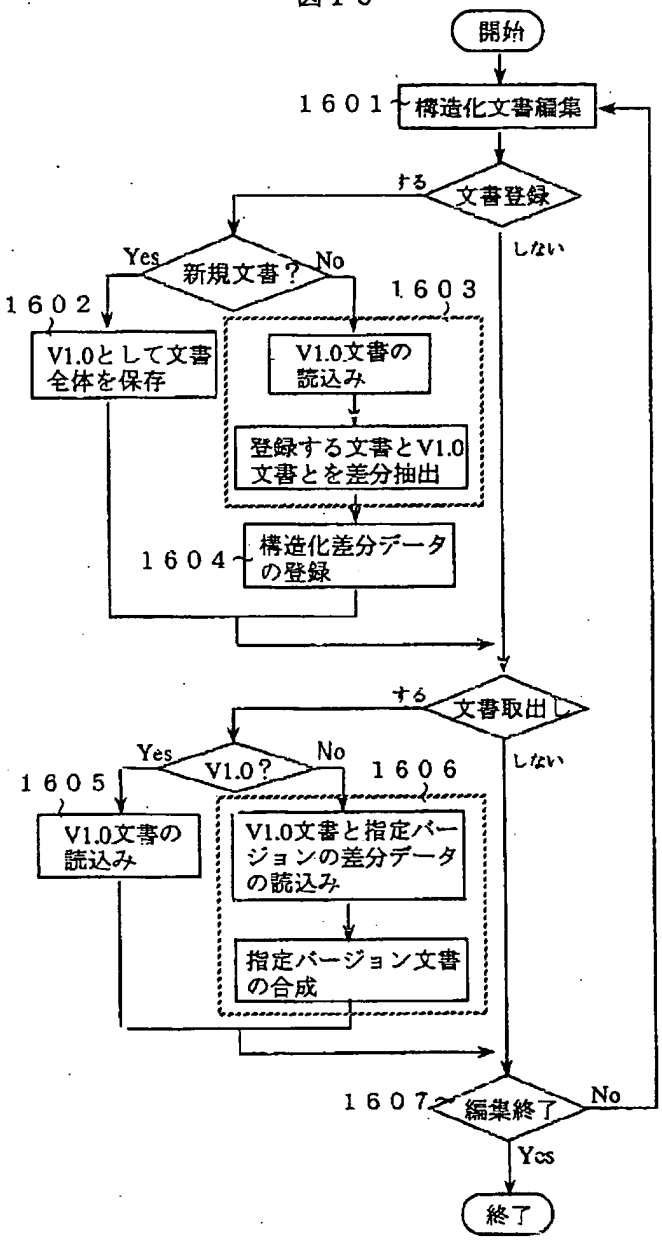
文書版管理プログラム
文書取出プログラム　1505
表示・編集プログラム　1508
文書登録プログラム　1506
構造化差分データ
差分抽出プログラム　1507

1508　1501
2次記憶装置　CPU
BUS
1502

BNSDOCID: <JP_____409319632A__I_>

【図16】

図16



【図17】

図17

```
<論文>
  <著者名>平成太郎</著者名>
  <章>
    <章番号>第１章</章番号>
    構造化文書とは？SGMLとは？
  </章>
  <章>
    <章番号>第２章</章番号>
    構造化文書の差分抽出方式
  </章>
</論文>
```

1701

(a) 構造化文書b（V1.0）

```
<論文>
  <著者名>昭和次郎</著者名>
  <所属>XYZ大学</所属>
  <章>
    <章番号>第１章</章番号>
    構造化文書とは？
  </章>
  <章>
    <章番号>第２章</章番号>
    構造化文書の差分抽出方式
  </章>
</論文>
```

1702

(b) 構造化文書b'（V2.0）

【図19】

図19

```
<論文>
  <著者名>
    <変更前>平成太郎</変更前>
    <変更後>昭和次郎</変更後>
  </著者名>
  <所属 difftag="挿入">XYZ大学</所属>
  <章>
    <章番号>第１章</章番号>
    構造化文書とは？<削除>SGMLとは？</削除>
  </章>
  <章>
    <章番号>第２章</章番号>
    構造化文書の差分抽出方式
  </章>
</論文>
```

1901

↓

```
<論文>
  <著者名>昭和次郎</著者名>
  <所属>XYZ大学</所属>
  <章>
    <章番号>第１章</章番号>
    構造化文書とは？
  </章>
  <章>
    <章番号>第２章</章番号>
    構造化文書の差分抽出方式
  </章>
</論文>
```

1902

BNSDOCID: <JP_____409319632A_I_>

フロントページの続き

(72)発明者　伊藤　泰樹
神奈川県横浜市戸塚区戸塚町5030番地　株式会社日立製作所ソフトウェア開発本部内

BNSDOCID: <JP_____409319632A__I_>